International Journal of Trend in Scientific Research and Development (IJTSRD)
Volume 5 Issue 4, May-June 2021 Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

An Extensive Review on Generative Adversarial Networks (GAN's)

Atharva Chitnavis[1], Yogeshchandra Puranik[2]

[1]PG Student, [2]Assistant Professor,
[1,2]Department of (MCA) P.E.S.'s Modern College of Engineering, Pune, Maharashtra, India

ABSTRACT

This paper is to provide a high-level understanding of Generative Adversarial Networks. This paper will be covering the working of GAN's by explaining the background idea of the framework, types of GAN's in the industry, it's advantages and disadvantages, history of how GAN's are developed and enhanced along the timeline and some applications where GAN's outperforms themselves.

How to cite this paper*:* Atharva Chitnavis | Yogeshchandra Puranik "An Extensive Review on Generative Adversarial Networks (GAN's)" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-5 | Issue-4, June 2021, pp.778-782, URL: www.ijtsrd.com/papers/ijtsrd42357.pdf

IJTSRD42357

1. INTRODUCTION

In the past many years there have been enormous enhancements in the field of Deep Learning. Deep learning is a branch of Machine Learning which deploys algorithms and imitates the thinking process of a human brain. This is done by embedding layers of Neural Networks together to carry out tasks such as Human speech, Object recognitions, etc. Feature extraction from the provided data is the main goal of Deep Learning. Artificial Neural Networks (ANN) is a sub-branch of Deep Learning. They are mimicked from the structure of neurons in our brain which are connected to each other and transmit signals based on the input they receive from the previous neuron. In 1951, Narvin Minsky made the first Artificial Neural Network while working at Princeton and since then there has been a substantial increase in the power on ANN's due to high computational processors. Computer vision uses highly complicated Neural Networks depending on the framework. Computer vision is a field of Neural Networks where we have to make the computer see the world which we perceive as pictures in the form of vector arrays as independent pixels. Learning feature representation from large data sets of unlabelled data has been a highly active area of research. Computer vision has the leverage of having large volumes of unlabelled datasets of videos and images available to be trained. Generative Adversarial Networks (GAN's) is a machine learning framework designed by Ian Goodfellow with the main focus of collecting data and creating new unseen noise data from the trained model. GAN's use two different neural networks in order to predict outcomes. The first NN is the Generator Model and the second NN is the Discriminator Model. Having these two models simultaneously has given GAN's a certain edge over the rest of the framework in filtering out fake data from the whole dataset. In order to make a GAN very effective we have to find a perfect balance between both the models so that the second model is not masking the output of the first model. GAN's are used nowadays very widely for computer vision-based tasks for accurate predictions and better results. Applications GAN's extends are (Generate Photographs of Human Faces, Super Resolution, 3D Object Generation, etc).

2. MODELS IN GAN's

2.1. GENERATIVE MODELS

Generative models have gained a lot of popularity in recent years. A generative model is used to create fresh new instances and fetch them forward. A generative model could create new videos, photos, or any kind of noise. This noise data can also be used to fill out the missing data or predict the missing data. Types of framework which used generative models are as follows. Generative models study the joint probability P(x,y).

	Y=0	Y=1
X=0	1/2	0
X=1	1/4	1/4

2.1.1. Bayesians Network
Bayesian's network is a generative probabilistic model which we can use effectively to represent random variables. This model works with the help of two main parts structure and parameter. The structure is the acyclic graph and parameters consist of probabilities within each node. Based on the output of these two parts the model predicts final probabilities and generates final outputs.

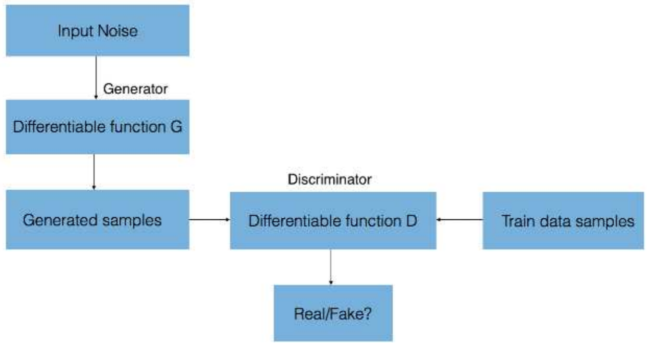
2.1.2. Gaussian Model
This model assumes that all data points are a mixture of finite Gaussians distribution. These points are then further used to predict outcomes such as biometric systems. The two main parts of this model are data points and equiprobability. Real life application of this model can be clustering iris databases.

2.2. Discriminative Model
Discriminative models use logistic regression techniques to discriminate between multiple categories. The main task is to train a model such that it could categorize the dataset by the features it receives. These are highly used in statistical classification in supervised learning. These models are also known as conditional models. These models support vector machines, decision trees and random forest. Discriminative model studies the joint probability of P(x|y) i.e. It predicts the probability of y targets when given x.

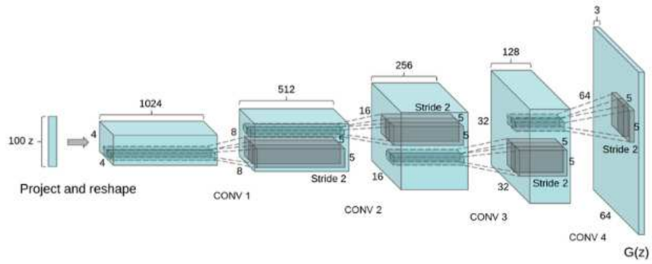
	Y=0	Y=1
X=0	1	0
X=1	1/2	1/2

3. GAN IS A TWO PLAYER GAME
GAN networks stand out from the other generative networks because it depends on two neural networks within itself for predicting outputs. The first network is a generative model. It provides GAN with large numbers of noise data. This data can be anything from images, videos, voice etc. This data is then used to train the second neural network which is the discriminator model. Discriminator model is to be trained to classify the given data from the generator model in correct slots. The discriminator should be able to tell whether the receiving data is correct or incorrect data. We have to find a balance between the generator and the discriminator such that the data is successfully classified. A classic example of GAN's is (Let's assume we have an Art thief whose work is to replicate original art work which will act as a generator model in our framework. Now we have an art inspector who can differentiate between real and fake copies of art work which will act as a discriminator model in our framework.) The basic idea behind GAN's is the same where two models will work together where one is fetching real and fake data and other is classifying the data in the categories. Larger the data set the better the discriminator model keeps growing. During the training process, weights and biases are adjusted through backpropagating weights until the discriminator learns to distinguish between real and fake images. The generator gets this feedback from the discriminator and uses it to produce more real images. The discriminator model is a convolutional neural network while the generator is a deconvolutional network.

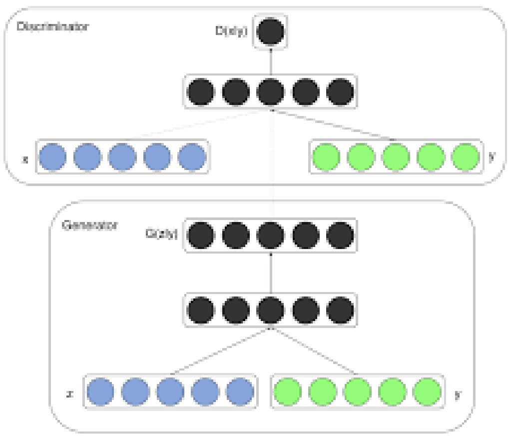
4. DIFFERENT TYPES OF GAN'S
4.1. Basic GAN's
This framework has two neural networks generator and the discriminator. Generator generates real and fake images and fetches them to the discriminator for further classification and training, while discriminator uses those images to classify them as real and fake to make the model stronger and efficient.



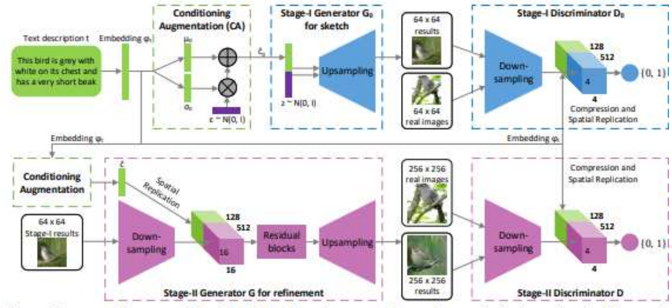
4.2. Deep Convolutional GAN's (DCGAN)
Convolutional networks can take image input and then extract specific important features from the image and are also able to differentiate between them. Rather than implementing hardly encoded filters on the image, convolutional networks are used to learn those filters automatically and filter out important aspects in the image. Similarly, DCGAN's are image versions of GAN's. DCGAN's use convolutional layers, here we replace max polling with convolutional strides, transposed convolution for up sampling, use ReLU in the generator model and Leaky ReLU in the discriminator model.



4.3. Conditional GAN's
CGAN is an update over the basic GAN. In GAN's, the generator model generates an image by plucking it from a very large dataset. These result in very random images which have no relation with our application. We can make the image generation conditional if applied to a class label to generate a specific type of image.

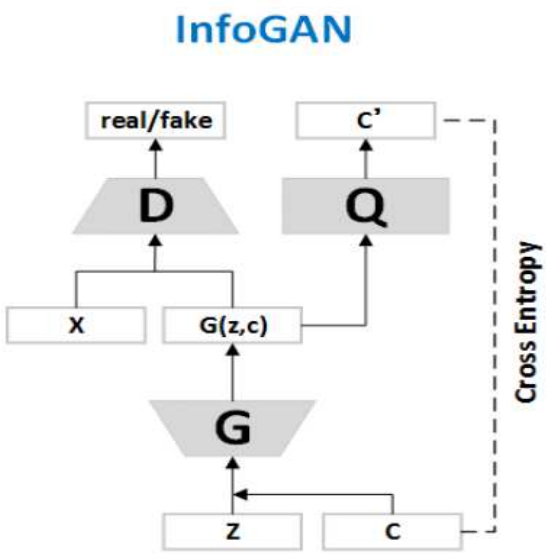


4.4. Stack GAN
The work of GAN's was to generate output models by filtering images from the dataset. This moto of GAN's was used and was implemented by applying text to image conversion. In Stack GAN the user provides a text description of the output image which is required. Then the stack GAN works in two stages. The first stage applies conditional augmentation on the text parameters and stacks images of 64x64 and upsamples it in the generator. These images are sent to the discriminator after down-sampling them. This result goes to the second stage where it is down sampled again to convert them into residual blocks resulting in a final generator output of 256x256 images, then finally the last discriminator down samples it again gets the final image output.

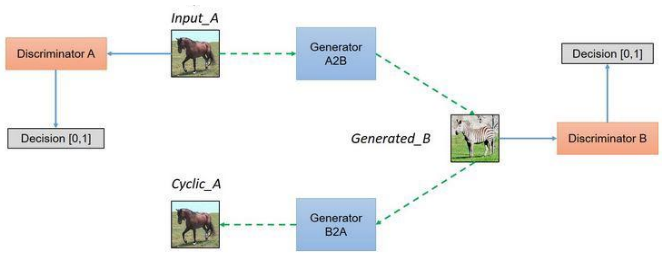


Applications of Stack GAN are (Comic creation, Art Creation, Movie Creation, High Quality Image Generation, etc.)

4.5. Info GAN
This is an improvement on GAN's and also is a contrast to the CGAN's where we use labels to filter the dataset. Info GAN is an unsupervised learning technique. Info GAN implements information theory from statistics in the GAN framework. Information theory suggests that there is a high volume of information in an unlikely event compared to a likely event. Info GAN is able to learn disentangled representations of image in an unsupervised manner. This model is used when the dataset is not labelled and is highly complex.



4.6. Cycle GAN's
Image-image translation is generating a newly synthesised version of a provided image with modified specifications. E.g. A horse image to a zebra image. In order to work perfectly they need a large dataset of paired examples which are highly expensive and difficult to prepare. To overcome this problem, we use Cycle GAN which involves automatic training of image-image translation models without needing paired examples. The model receives images from source and target domain which don't need to be related in any way. Individual images are selected from these two domains and the required features are extracted from them. Then these features are combined to make a third translated image.



4.7. Discover Cross-Domain Relations with GAN (Disco GAN's)
It is easy for humans to recognize relations between different things from multiple domains easily without the help of external supervision. This is a difficult task for machines to achieve as they can automatically relate two things. Disco GAN are very closely related to Cycle GAN. Disco GAN uses two GAN networks inside itself that maps each domain to its counterpart domain. The reconstruction loss is used to verify how well the image is translated from one Domain 1 to Domain 2 and vice versa. Each domain of Disco GAN has its separate reconstruction losses which makes a huge impact on cross domain reconstruction.

5. IMPLEMENTING GAN'S
5.1. Variable Settings
As GAN's are highly used in the application of image generation we will be using it in a DCGAN. Firstly, we will set the parameters needed for the GAN such as (dataset root, threads to load data, batch size, image size, colour channels to use, length of vector, epoch numbers, setting learning rate). Next is to set weight. The weight_init function takes the previous model as input and batch normalization, convolutional-transpose.

5.2. Generator Setup
Generator is designed to map the latent space vector to data space. Since our data is in the form of an image, we are converting them into an RBG vector with the same dimension as the image. The output of the generator is sent to a tanh function to normalize it in output range of [-1,1].

5.3. Discriminator Setup
Discriminator is a binary classifier network that will take an image as an input and give a scalar probability output which will tell us if the input image is a real or a fake. The input image from the generator goes through Conv2d, BatchNorm2d, Leaky ReLU layers and outputs the final output through a sigmoid function

5.4. Loss Functions and Optimizers
Once the generator and discriminator are set, we can specify how they can learn from their loss. We use a Binary Cross Entropy function from PyTorch and then optimize the output Generator and Discriminator using ADAMS optimizer.

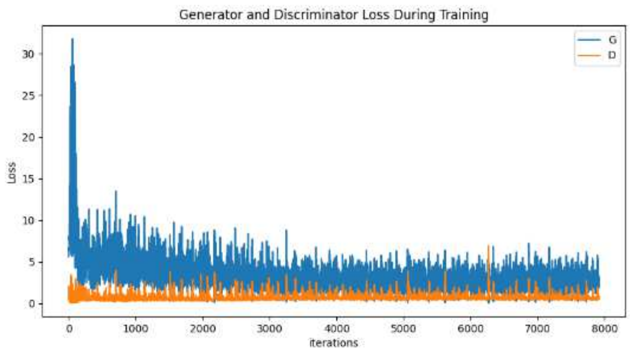
5.5. Training GAN
After both the neural nets have been set, we need to train our model in two parts. GAN's training is a tricky part because it can fail

Due to wrong parameters and won't give much of an explanation for the collapse.

5.5.1. Training the discriminator
The goal of the discriminator is to classify the given input in real or fake with high probabilities. Firstly, we need to train it with a batch of real samples from the training dataset and pass it to the discriminator, then calculate the gradient loss in a backward pass. Then we need to do the same process with the fake samples from the dataset. Now the gradient pass from both the generator and discriminator can help us set a step for the discriminator optimizer.

5.5.2. Training the generator
The work of the generator is to develop high quality fakes. We can achieve this with the help of the gradient pass received from the discriminator.

5.6. Results
Given below is the graph developed by implementing DCGAN for generating random images from the dataset we provide. We generate 2 different outputs as (Discriminator and Generator losses along with Generator's output with fixed noise). In the graph we can see that the generator loss goes on decreasing as the iteration goes on increasing. Lower the loss closer is the model to generate images.



6. HISTORY OF GAN'S
Generative Adversarial Networks was a framework developed by Ian Good fellow in the final year of his PhD in 2014. This idea was implemented further in 2017 for enhanced focus on realistic texture rather than pixel accuracy which helped in generating higher quality images on even high levels of magnifications. In the year 2017 GAN generated a first virtual human face which was displayed in 2018 at the Grand Palais. An innovative solution named "Creative Adversarial Network" was developed and sold in 2018 which was able to generate appealing high quality abstract paintings. In 2019 GAN produced a first human talking video by generating each and every frame on its own given only a single photo of that person. In 2020 Nvidia taught an AI system (GameGAN) to recreate the complete game of PAC-MAN by just simply watching it being played. GAN has developed exponentially in a short time due to the vast applications it can be applied in and also the idea of implementing multiple neural nets in such a framework. It gave it an edge especially in the computer vision field to generate images efficiently.

7. Why Generative Adversarial Networks
In the 21th century machine learning has influenced many areas in science, commerce and arts. All the way from diagnosing skin diseases, generating abstract arts, to enhancing credit systems we are able to implement machine learning algorithms everywhere. One of the most challenging part is to fool the existing algorithm of neural networks by adding noise data into original datasets. To tackle the problem of Deepfake data we use neural networks such as GAN's Having two neural nets inside the framework helps to keep up with the problem of filtering out noise from the dataset easily solving the problem of Deepfakes.

8. APPLICATIONS OF GAN'S
8.1. Image Processing
8.1.1. Image Dataset Generations
GAN's very first use was generating datasets of images from the available sample data. Such as (Adding glasses to a face image for a face with no glass). We can generate many such types of datasets using GAN's

8.1.2. Super Resolution
We have used GAN's to generate images with much higher resolution than the original image so that they won loose detail due to magnification.

8.1.3. Face Generations
Generating human face images and videos by collecting information from the available dataset was one pinnacle achieved with the help of GAN's.

8.1.4. Text to image translation
This was a useful functionality now possible with the help of GAN. We can input a lie of text and the GAN network will generate an image based on the requirement entered by the user.

8.2. Speech Generation
8.2.1. Music Generation
GAN was able to generate melodious audio on its own by knowing what kind of music humans like. By analysing data of many musical libraries, it was able to train the network and generate audio based on the discriminator.

8.2.2. Speech Generation
We were also able to achieve complete speech generation with help of GAN from scratch. We can feed topics to the network as a label and the model returns a complete speech in return.

9. ADVANTAGES AND DISADVANTAGES OF GANS

9.1. Advantages of GAN's

A. GAN can generate high volumes of unseen data in form of images, audio, video, text.

B. They don't need any kind of labelled data for generation or working of the network. They come under unsupervised learning.

C. GAN's are able to learn highly complicated datasets with various data distributions.

9.2. Disadvantages of GAN's

A. During the gradient backflow from final to the first layer the slope keeps on getting smaller. At times the slopes are so small that the starting layers learn very slowly or sometimes even stop learning.

B. The model collapses due to wrong hyper parameters and does not provide with much of an explanation for developer to solve the problem

10. Conclusion

This paper presents an extensive view on GAN's, types of GAN's, Implementation, History, Advantages and Disadvantages and its applications. I believe this paper will help the reader get an in-depth understanding of GAN's and its working.

REFERENCES

[1] Ian J. Goodfellow, Jean Pouget-Abadie_, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozairy, Aaron Courville, Yoshua Bengio. "Generative Adversarial Networks"

[2] Mehdi Mirza "Conditional Generative Adversarial Networks"

[3] Alec Radford & Luke Metz" Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks"

[4] https://pytorch.org/tutorials/beginner/dcgan_faces_tutorial.html

[5] https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/